**Contratti per l’UC “Gestire Eventi”**

**Pre-condizione generale:** l’attore è identificato con un’istanza *org* di Organizzatore

# **1. creaEvento(dataInizio: Data, cliente: Testo, nome: Testo):**

**Pre-condizioni:** ---

**Post-condizioni:** se dataInizio non è ancora passata:

* l’istanza *me* di Macro-Evento viene creata
* viene creata l’associazione **gestisce** tra *org* e *me*
* è stata creata l’istanza *ev* di Evento che **è ricorrenza di** *me*
* *ev.*data = dataInizio
* *me.*cliente = cliente
* *me.*nome = nome

# **1a.1 apriEvento(evento: Evento):**

**Pre-condizioni:** ---

**Post-condizioni:** Se evento.stato = programmato/attivo e *org* **gestisce** Macro-Evento *me* di cui evento **è ricorrenza**

# **1a.2 annullaEvento(strategia: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*

**Post-condizioni:** se l’evento *evento.*stato è attivo:

* *evento*.stato = annullato
* per ogni Servizio *s* in cui **è suddiviso** *evento:*
  + ogni istanza *is* Incarico di Servizio **deciso per** s viene eliminata
  + ogni *ic* Incarico di Cucina **distribuito su** *c* Compito**organizzato** dall’istanza *foglio* di Foglio Riepilogativo **associato** a *s*, viene eliminato
* [se strategia = “questo e successivi”] le modifiche sopra effettuate sono applicate anche ad ogni Evento *evento2* che **è ricorrenza di** *me*,che abbia *evento2*.stato = programmato e che *evento*.dataInizio <= *evento2*.dataInizio
* [se strategia = “tutti”] le modifiche sopra effettuate sono applicate anche ad ogni Evento *evento2* che **è ricorrenza di** *me* eche abbia *evento2*.stato = programmato

# **1a.2a.1 terminaEvento():**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento*

**Post-condizioni:** se l’evento *evento.*stato è attivo:

* *evento*.stato = terminato

# 

# **1a.2b.1 eliminaEvento(strategia: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*

**Post-condizioni:** se *evento.*stato = programmato:

* ogni Servizio *s* in cui **è suddiviso** *evento s* viene eliminato e*:*
  + ogni istanza *is* Incarico di Servizio **deciso per** *s* viene eliminata
  + il *foglio* Foglio Riepilogativo **associato a**  *s* viene eliminato e:
    - ogni istanza Compito *c* che *foglio* **organizza** viene eliminata e:
      * ogni *ic* Incarico di Cucina **distribuito su** *c* viene eliminato
* l’istanza *evento* viene eliminata
* [se strategia = “questo e successivi”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me, che abbia evento2.stato = programmato e che evento.dataInizio <= evento2.dataInizio
* [se strategia = “tutti”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me e che abbia evento2.stato = programmato

# **1b.1 apriEventoTerminatoAnnullato(evento: Evento):**

**Pre-condizioni:** ---

**Post-condizioni:** Se evento.stato = terminato/annullato e *org* **gestisce** Macro-Evento *me* di cui evento **è ricorrenza**

# **1b.2 gestisciNote(note: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*

**Post-condizioni:** *me.*note = note

**2. aggiungiNote(note: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*

**Post-condizioni:** *me.*note = note

# **3. visualizzaStatoCucina():**

**Pre-condizioni:** ---

**Post-condizioni:** ---

# **4. modificaDettagliEvento(dataInizio?: Data, cliente?: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*

**Post-condizioni:** se dataInizio non è ancora passata e se *ev.*stato è programmato:

* [se è specificata dataInizio] *evento*.dataInizio = dataInizio
* [se è specificato il cliente] *me*.cliente = cliente

**4a.1 modificaPeriodicità(eventoBase: Evento, periodicità: Periodicità, dataPartenza: Data):**

**Pre-condizioni:** eventoBase **è ricorrenza** di Macro-Evento *me*

**Post-condizioni:**

* se *me* **si ripete con** Periodicità *p,* questa viene sostituita con la nuova istanza periodicità e vengono create tante istanze *eventoFuturo* di Evento quanto specificato da periodicità.numeroOccorrenze con:
  + *eventoFuturo.*dataInizio=*eventoBase.dataInizio+*(*eventoBase.*offsetGiorniOccorrenza \* *eventoBase.*numeroOccorrenze)
  + per ogni istanza di Servizio *servizio* in cui *eventoBase* **è suddiviso**, viene creata una copia dell’istanza *servizio* che **suddivide** *eventoFuturo*, assieme a tutte le Proposte Di Modifica che **modificano** *servizio*
  + *eventoFuturo*  viene **assegnato** all’istanza Chef *c* a cui  *eventoBase* era **assegnato**
  + la relazione **è ricorrenza di** tra *eventoFuturo* e *me* è creata
  + *eventoFuturo*.stato = programmato
* Inoltre, tutte le istanze di Evento *vecchioEvento* che **sono ricorrenza di** *me*, che non corrispondono più alla nuova periodicità e hanno *vecchioEvento*.stato = Programmato vengono rimosse.

# **5. aggiungiServizio(nome: Testo, offsetInizio: Numero, rangeOrario: RangeOrario, numeroCommensali: Numero, luogo: Testo, dettagli: Testo, strategia: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*

**Post-condizioni:** se offesetInizio non identifica un giorno già passato:

* Viene creata una nuova istanza *s* Servizio con:
* s.offsetInizio = offsetInizio
* *s.*orario = rangeOrario
* *s.*dettagli = detttagli
* *s.*luogo = luogo
* *s.*nome = nome
* *s.*numeroCommensali = numeroCommensali
* la relazione **è suddiviso in** tra s ed *evento* viene creata
* [se strategia = “questo e successivi”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me, che abbia evento2.stato = programmato e che evento.dataInizio <= evento2.dataInizio
* [se strategia = “tutti”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me e che abbia evento2.stato = programmato

**5a.1 annullaServizio (servizio: Servizio, strategia: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me,* *evento* **è suddiviso in** servizio

**Post-condizioni:** se servizio.stato non è annullato o terminato:

* per ogni Personale *p* **deciso per** servizio*,* ogni istanza *is* Incarico di Servizio in cui *p* **serve** viene eliminata
* ogni *ic* Incarico di Cucina **distribuito su** *c* Compito**organizzato** dall’istanza *foglio* di Foglio Riepilogativo **associato** a servizio, viene eliminato.
* [se strategia = “questo e successivi”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me, che abbia evento2.stato = programmato e che evento.dataInizio <= evento2.dataInizio
* [se strategia = “tutti”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me e che abbia evento2.stato = programmato

**5b.1 eliminaServizio (servizio: Servizio, strategia: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me,*  *evento* **è suddiviso in** servizio

**Post-condizioni:** se servizio.stato non è confermato, annullato o terminato:

* per ogni Personale *p* **deciso per** *s,* ogni istanza *is* Incarico di Servizio in cui *p* **serve** viene eliminata
* il *foglio* Foglio Riepilogativo **associato a**  *s* viene eliminato e:
  + ogni istanza Compito *c* che *foglio* **organizza** viene eliminata e:
    - ogni *ic* Incarico di Cucina **distribuito su** *c* viene eliminato
* servizio viene eliminato
* [se strategia = “questo e successivi”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me, che abbia evento2.stato = programmato e che evento.dataInizio <= evento2.dataInizio
* [se strategia = “tutti”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me e che abbia evento2.stato = programmato

# **5c.1 modificaDettagliServizio(servizio:Servizio, nome?: Testo, offsetInizio?: Numero, rangeOrario?: RangeOrario, numeroCommensali?: Numero, luogo?: Testo, dettagli?: Testo, strategia: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*; *evento* **è suddiviso in** servizio

**Post-condizioni:**  se servizio.stato non è confermato, annullato o terminato:

* [se nome è specificato] servizio*.*nome = nome
* [se offsetInizio è specificato] servizio.offsetInizio = offsetInizio
* [se rangeOrario è specificato] servizio*.*orario = rangeOrario
* [se detttagli è specificato] servizio*.*dettagli = detttagli
* [se luogo è specificato] servizio*.*luogo = luogo
* [se numeroCommensali è specificato] servizio*.*numeroCommensali = numeroCommensali
* [se strategia = “questo e successivi”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me, che abbia evento2.stato = programmato e che evento.dataInizio <= evento2.dataInizio
* [se strategia = “tutti”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me e che abbia evento2.stato = programmato

# 

# **6. assegnaChef(chef: Chef, strategia: Testo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*

**Post-condizioni:**

* chef viene **assegnato in** *evento*
* [se strategia = “questo e successivi”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me, che abbia evento2.stato = programmato e che evento.dataInizio <= evento2.dataInizio
* [se strategia = “tutti”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me e che abbia evento2.stato = programmato

# **7. confermaMenuServizio(servizio: Servizio):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* **che è suddiviso in** servizio

**Post-condizioni:** Se servizio.stato = in preparazione e un Menu **è scelto per** servizio:

* servizio.stato = confermato

# **7a.1 proponiModificaMenu(servizio: Servizio, menu: Menu, proposta: Proposta di Modifica, strategia: Testo):**

**Pre-condizioni:** menu **è scelto per** servizio; è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*; *evento* **è suddiviso in** servizio

**Post-condizioni:** Se servizio.stato = in preparazione:

* Viene creata l’associazione **modifica** tra proposta e servizio
* [se strategia = “questo e successivi”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me, che abbia evento2.stato = programmato e che evento.dataInizio <= evento2.dataInizio
* [se strategia = “tutti”] le modifiche sopra effettuate sono applicate anche ad ogni Evento evento2 che è ricorrenza di me e che abbia evento2.stato = programmato

# 

# **8. assegnaPersonale(dipendente: Personale, turnoServizio: TurnoServizio, ruolo?: Ruolo):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*; *evento* **è suddiviso in** *servizio* Servizio

**Post-condizioni:** se *servizio*.stato non è annullato o terminato:

* viene creata l’associazione **deciso per** tra *servizio* e dipendente
* viene creato un *is*  Incarico di Servizio e:
  + [se ruolo è specificato] *ic.*ruolo = ruolo
  + viene creata l’associazione **previsto in** tra *is* e turnoServizio
  + viene creata l’associazione **serve in** tra dipendente e *is*

# 

# **8a. rimuoviPersonale(dipendente: Personale, turno: TurnoServizio):**

**Pre-condizioni:** è in corso la visualizzazione di un Evento *evento* che **è ricorrenza del** Macro-Evento *me*; *evento* **è suddiviso in** *servizio* Servizio

**Post-condizioni:** se *servizio*.stato non è annullato o terminato:

* per ogni *personale* Personale che è **deciso per** *servizio*
  + viene eliminato ogni *is* Incarico di Servizio in cui *personale* **serve**